Analysis and Implementation of Low Power Cyclic Redundancy Check

Project Proposal

Ankit Gupta and Anup Shrinivasan ECE 6332

Introduction:

Cyclic redundancy check (CRC) is an error detecting code designed to detect changes in transmitted data and is used in digital networks and storage devices. CRC's are easy to implement in binary hardware, analyze mathematically and are used to detect errors due to noise in the transmission channel. The message is treated as a polynomial in GF(2). For example, the message 1111001 is represented as a polynomial as $x^6+x^5+x^4+x^3+1$. Specification of a CRC code also requires a pre-defined generator polynomial. The sender and receiver agree on a certain fixed polynomial. The remainder is calculated as by dividing the GF(2) polynomial by the generator polynomial to generate a reliable check sum. Division of polynomials over GF(2) can be done in much the same way as long division of polynomials over the integers. This can be implemented using an exclusive OR gate. At the receiver, the integrity of the data is checked similarly. The data is said to be valid if the check sum at the receiver yields all zeros.

Initial Project Progress:

- *Project page on Wiki:*

  https://venividiwiki.ee.virginia.edu/mediawiki/index.php/ClassECE6332Fall11GroupPages

- Implemented CRC circuit using LSFR and verified the functionality of the circuit using Cadence.
- LSFR was implemented as a master slave topology with a synchronous reset using multiplexors.
- The XOR gates were implemented using transmission gates.
- Implemented the CRC circuit using TSPC registers in two different topologies.

Analysis and Implementation of Low Power Cyclic Redundancy Check

Project Proposal

Ankit Gupta and Anup Shrinivasan ECE 6332

Project Goals:

In this project, we motivate the concept of comparing CRC circuits based on their energy-delay trade-offs and analyze the energy-delay space of a sample linear CRC implementation. A methodology is applied to minimizing the energy of the CRC circuit under delay constraints. Impact of various design choices, including the Linear Feedback Shift Registers (LFSR) topology, tradeoff between acceptable glitches and the use of buffers to reduce glitching effect, transistor sizing, etc are analyzed in the energy-delay space and verified. The result of the verification will be demonstrated on a design resulting in the most energy efficient CRC implementation.

Design Component:

The design component of the project involved implementing a cyclic redundancy check circuit using Linear Feedback Shift Registers (LFSR). We use a generator polynomial $G(x)=x^6+x^5+x^4+x^3+1$. The CRC-6 is a method of performance monitoring that is contained within the F-bit position of frames 2, 6, 10, 14 18 and 22 of every multiframe in Synchronous Frame structures used at 1544, 6312, 2048, 8448 AND 44 736 kbit/s Hierarchical levels. We start off by designing transmission gates, XOR gates, Master slave flip flop implemented as a linear feedback shift register and Multiplexors (for synchronous reset) using the sizing constraints. Then we look into TSPC flip flop for implementation of the register (low power). The next step would be to explore reducing the total power and glitching in the CRC implementation by designing a power efficient LFSR topology.

Research Component:

We plan on investigating the circuit for low power functionality of the CRC circuit. Further, we hope to minimize the total power usage of the circuit using different flip flop topologies. After our initial

progress for the design review, we implemented the CRC circuit using TSPC flip flop and noticed increased glitching in the output as shown in Figure 3.

Yang et al [1] explored the TSPC flip flops and proposed a register topology with minimal glitching. We studied the circuit and implemented the design in the CRC. The overall circuit now comprises of shift registers built using the design in [1] with noticeable glitch reduction, which is shown in our simulations in Figure 2. The register topology proposed in [1] uses half the number of transistors to build the overall flip flop, as compared to our initial design. We plan on analyzing the low power functionality of the overall CRC circuit based on the Pareto curve (energy vs. delay).

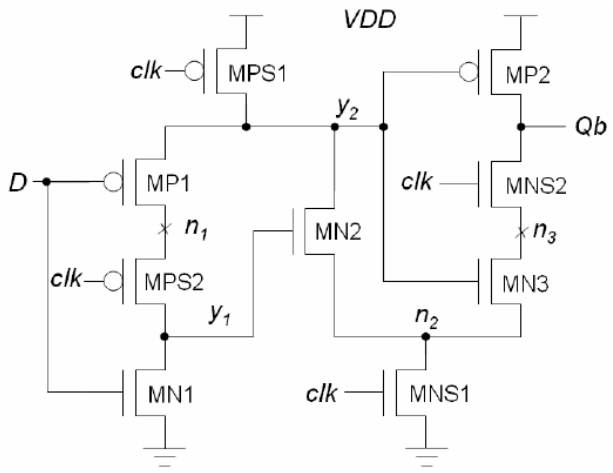


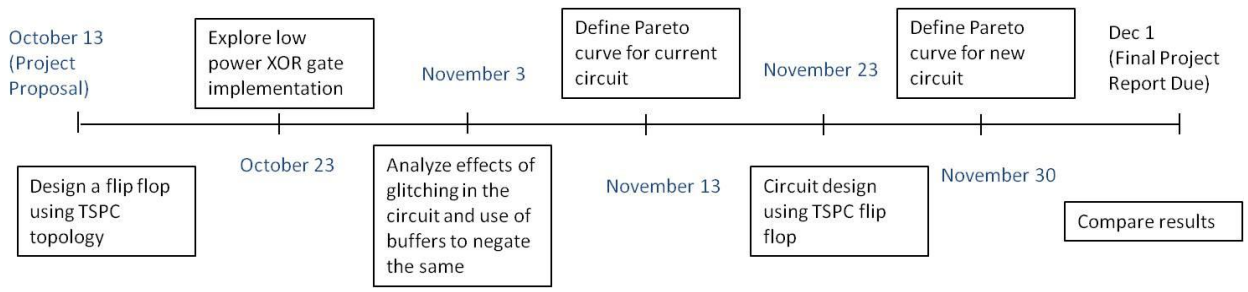
Figure 1: Proposed D-Flip Flop for glitch reduction [1]

Analysis and Implementation of Low Power Cyclic Redundancy Check

Project Proposal

Ankit Gupta and Anup Shrinivasan ECE 6332

Task Timeline:



Task Breakdown:

The aim of the project is two fold; identify glitching factors and eliminate the same, and analyze the low power functionality of the overall circuit. Although we will be doing both in tandem as a team, our individual focus includes,

- ***Ankit Gupta:***
  - Analyze effects of glitching in the circuit.
- ***Anup Shrinivasan:***
  - Circuit design using TSPC flip flop and analysis of low power functionality of the circuit.

References:

[1] Sung Hyun Yang, Younggap You, "A New Dynamic Flip Flop Aiming at Glitch and Charge Sharing Free" IEICE Trans Electron, Vol – E86-C, No. 3, March 2003.

[2] Zlatanovici, R, Sean Kao, Nikolic, "Energy–Delay Optimization of 64-Bit Carry-Lookahead Adders With a 240 ps 90 nm CMOS Design Example", IEEE Journal of Solid-State Circuits, Feb. 2009

Analysis and Implementation of Low Power Cyclic Redundancy Check

Project Proposal

Ankit Gupta and Anup Shrinivasan ECE 6332

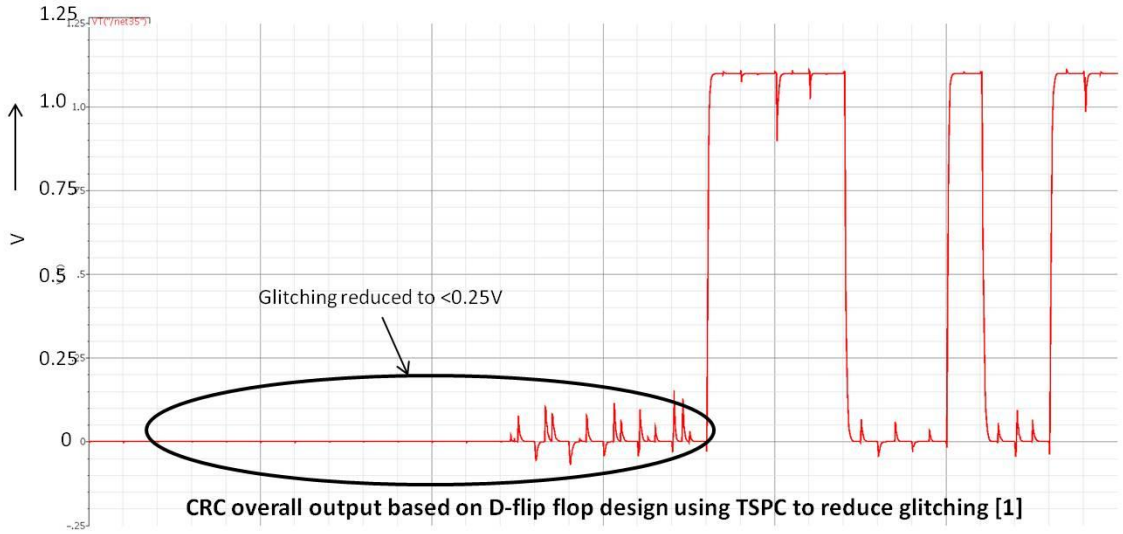


Figure 2: Glitch reduction due to topology discussed in [1]

Analysis and Implementation of Low Power Cyclic Redundancy Check

Project Proposal

Ankit Gupta and Anup Shrinivasan ECE 6332

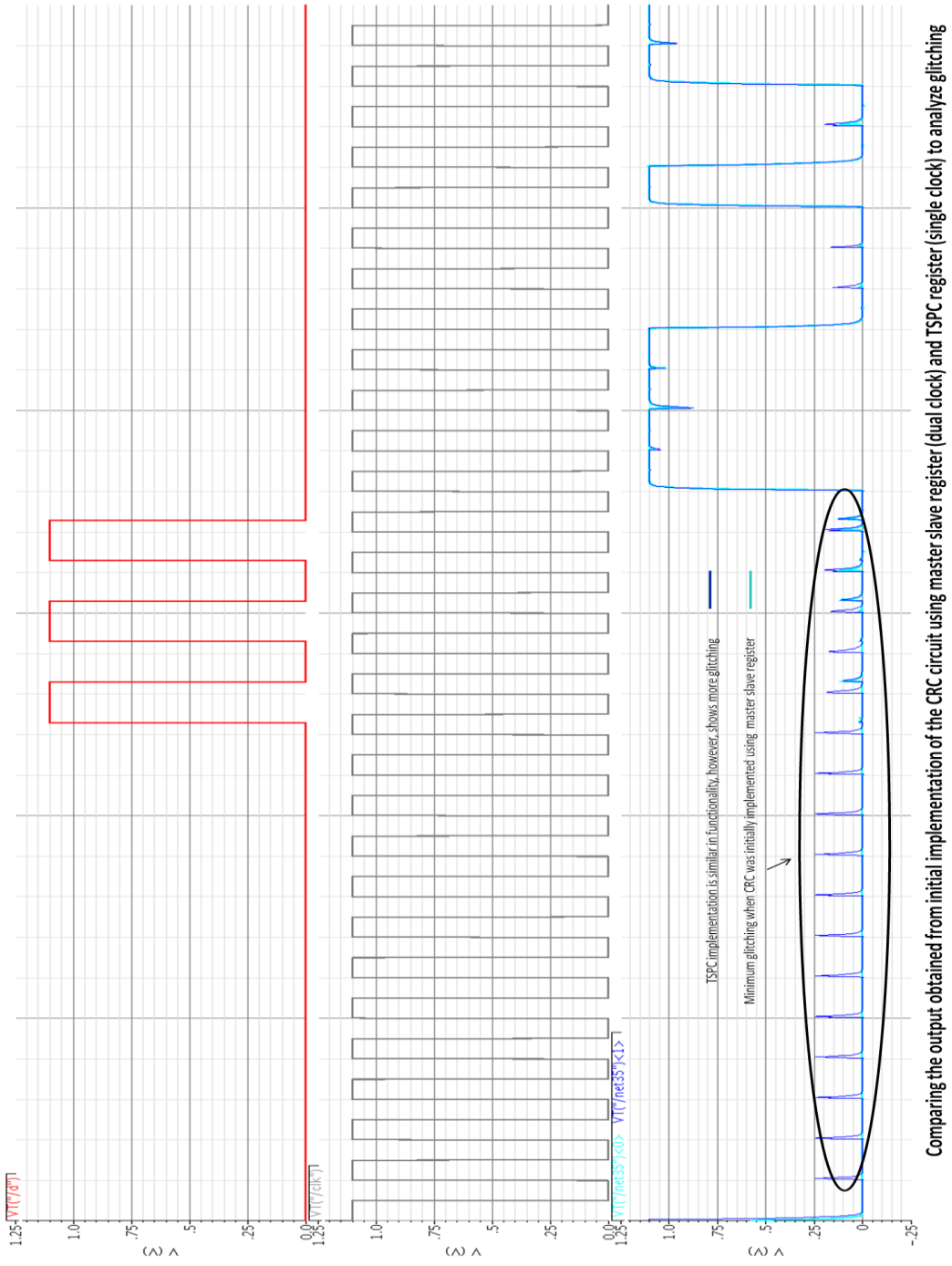


Figure 3: Glitching observed in the TSPC flip flop implementations